

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Total Number of Pages in This Submission

23

Application Number	10/015,301
Filing Date	12/12/2001
First Named Inventor	Matthias Stefan Bierbrauer
Art Unit	2136
Examiner Name	Pramila Parthasarathy

Attorney Docket Number

DE920000115US1

ENCLOSURES (Check all that apply)		
<input checked="" type="checkbox"/> Fee Transmittal Form <input type="checkbox"/> Fee Attached <input type="checkbox"/> Amendment/Reply <input type="checkbox"/> After Final <input type="checkbox"/> Affidavits/declaration(s) <input type="checkbox"/> Extension of Time Request <input type="checkbox"/> Express Abandonment Request <input type="checkbox"/> Information Disclosure Statement <input type="checkbox"/> Certified Copy of Priority Document(s) <input type="checkbox"/> Reply to Missing Parts/ Incomplete Application <input type="checkbox"/> Reply to Missing Parts under 37 CFR 1.52 or 1.53	<input type="checkbox"/> Drawing(s) <input type="checkbox"/> Licensing-related Papers <input type="checkbox"/> Petition <input type="checkbox"/> Petition to Convert to a Provisional Application <input type="checkbox"/> Power of Attorney, Revocation <input type="checkbox"/> Change of Correspondence Address <input type="checkbox"/> Terminal Disclaimer <input type="checkbox"/> Request for Refund <input type="checkbox"/> CD, Number of CD(s) _____ <input type="checkbox"/> Landscape Table on CD	<input type="checkbox"/> After Allowance Communication to TC <input type="checkbox"/> Appeal Communication to Board of Appeals and Interferences <input checked="" type="checkbox"/> Appeal Communication to TC (Appeal Notice, Brief, Reply Brief) <input type="checkbox"/> Proprietary Information <input type="checkbox"/> Status Letter <input checked="" type="checkbox"/> Other Enclosure(s) (please Identify below): 1) Return Postcard
Remarks		

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm Name	Winstead Sechrest & Minick, P.C.		
Signature			
Printed name	Robert A. Voigt, Jr.		
Date	February 28, 2006	Reg. No.	47,159

CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below:

Signature

Typed or printed name

Wanda Kellar

Date February 28, 2006

This collection of information is required by 37 CFR 1.5. The information is required to obtain or retain a benefit by the public which is to file (and by the USPTO to process) an application. Confidentiality is governed by 35 U.S.C. 122 and 37 CFR 1.11 and 1.14. This collection is estimated to 2 hours to complete, including gathering, preparing, and submitting the completed application form to the USPTO. Time will vary depending upon the individual case. Any comments on the amount of time you require to complete this form and/or suggestions for reducing this burden, should be sent to the Chief Information Officer, U.S. Patent and Trademark Office, U.S. Department of Commerce, P.O. Box 1450, Alexandria, VA 22313-1450. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.



- 1 -

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of: : Before the Examiner:
 Matthias Stefan Bierbrauer : Pramila Parthasarathy
 :
 Serial No.: 10/015,301 : Group Art Unit: 2136
 :
 Filing Date: December 12, 2001 :
 :
 Title: METHOD AND SYSTEM : IBM Corporation
 FOR AUTOMIZED AND : P.O. Box 12195
 SYNCHRONOUS EXECUTION OF : Dept. 9CCA, Bldg. 002-2
 CUSTOMIZED CODE ON OFF- : Research Triangle Park, NC 27709
 LOADED RETRIEVED :
 IN A DOCUMENT PROCESSING :
 SYSTEM :
 :

APPEAL BRIEF

Mail Stop Appeal Brief-Patents
 Commissioner for Patents
 P.O. Box 1450
 Alexandria, VA 22313-1450

I. REAL PARTY IN INTEREST

The real party in interest is International Business Machines, Inc., which is the assignee of the entire right, title and interest in the above-identified patent application.

CERTIFICATION UNDER 37 C.F.R. §1.8

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450, on February 28, 2006.

Signature

Wanda Kellar

(Printed name of person certifying)

03/06/2006 DEMMANU1 00000055 090461 10051301

01 FC:1402 500.00 DA

03/06/2006 DEMMANU1 00000076 090461 10015301

01 FC:1402 500.00 DA

II. RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences known to Appellants, Appellants' legal representative or assignee which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III. STATUS OF CLAIMS

Claims 1-7 are pending in the Application. Claims 1-7 stand rejected. Claims 1-7 are appealed.

IV. STATUS OF AMENDMENTS

Appellants have not submitted any amendments following receipt of the final rejection with a mailing date of July 28, 2005.

V. SUMMARY OF CLAIMED SUBJECT MATTER

In one embodiment of the present invention, a method for processing documents being off-loaded in a document processing system in which requests are executed by an off-loading process may comprise providing customized code for execution during the offloading process, and invoking the customized code when an off-loading request is issued synchronously to the off-loading process. Specification, p. 4, lines 1-14; Fig. 1, elements 101- 106; Claim 1. The method may be practiced in instances wherein the customized code is invoked synchronously to process the document before the document has been off-loaded from the document processing system. Specification, p. 4, lines 7-9; Fig. 1, elements 104- 105; Claim 2. The method may be practiced in instances wherein the customized code is invoked synchronously to process the document after the document has been off-loaded from the document processing system. Specification, p. 4, lines 10-12; Fig. 1, elements 105- 106; Claim 3. The method may be practiced in instances wherein the customized code forms an agent, the agent being invoked via a plug-in interface to

the document processing system. Specification, p. 5, lines 12-23; Fig. 1, elements 104, 106; Claim 4. The method may be practiced in instances wherein the customized code runs inside a plug-in architecture. Specification, p. 5, lines 12-23; Fig. 1, elements 104, 106; Claim 5.

In one embodiment of the present invention, a method for executing an agent comprising customized code relative to documents being archived from a document processing system may comprise the steps of selecting a document resident in the document processing system for archiving; creating an archiving request for the selected document to an archiving engine associated with the document processing system; invoking a pre-archiving agent, if any, on the selected document; when the pre-archiving agent is finished, archiving the selected document; invoking a post-archiving agent, if any, on the archived document; and when the post-archiving agent is finished, marking the archived document as 'archived'. Specification, p. 4 line 15 to p. 5, lines 12; Fig. 2; Claim 6.

In a computing environment, one embodiment of the present invention may be practiced as a system for processing documents being off-loaded in a document processing system in which requests are executed by an off-loading process, said system comprising: means for providing customized code for execution during the off-loading process; and means for invoking the customized code when an off-loading request is issued relative to a document in the document processing system synchronously to the off-loading process. Specification, p. 4, lines 1-14; Fig. 1, elements 101-106; Claim 7.

VI. GROUNDΣ OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1-7 stand rejected under 35 U.S.C. §102 as being anticipated by *Ferguson et al.*, U.S. Patent No. 6,810,404, hereinafter *Ferguson*.

VII. ARGUMENT**A. Rebuttal to the Examiner's Response**

In the Office Action mailed 07/28/2005, the Examiner expands her teachings while maintaining the previous grounds for rejection. *See* Office Action from 07/28/2005, p. 2, lines 7-11. Appellants respectfully traverse the Examiner's interpretation of *Ferguson* and assert that claims 1-7 are patentable over the reference.

A1. The Examiner's Misinterpretation of the Claim Language

Firstly, Appellants respectfully traverse the Examiner's overly broad interpretation of the terms "customized code" and "off-loading." *See* Office Action from 07/28/2005, p.3, line 19 to p. 4, line 2. The Examiner states that: "Applicant does not explicitly define or disclose 'customized code' or 'off-loading'." *See* Office Action from 07/28/2005, p.3, line 13. Appellants respectfully traverse and assert that these terms have been given sufficient meaning in the Specification and Drawings and from subject matter well known to one of ordinary skill in the art.

Regarding the term "customized code", the Examiner has asserted the meaning "as a program code for accomplishing a task," *see* Office Action from 07/28/2005, p.3, lines 19-20, for interpreting this term in claims 1-7. Appellants respectfully traverse the Examiner's interpretation and respectfully assert that her meaning renders the term "customized" *de facto* meaningless, as it simply excludes any program code that does not accomplish a task. However, it is well known to one of ordinary skill in the art that "customized code" refers to functionality that can be tailored within a development environment, using a programming language to generate, i.e., customize, a set of instructions, i.e., the code, which then may be compiled or interpreted, and then executed at run-time. Therefore, "customized code" inherently refers to a programmatic interface. By contrast, using a predefined user-interface at run-time to simply operate an executing program and enter values therein could not be construed as "customizing code," because this is not a programmatic interface, i.e., there is no change to the underlying code that has been compiled or

interpreted before being executed. Appellants respectfully assert that these concepts are well known and established in the art of computer programming and that the Examiner has provided no objective evidence or citation of a source for her unreasonably broad interpretation of "customized code."

Regarding the term "off-loading", the Examiner recites the meaning "'to assume part of the processing demand from another device', see Microsoft Computer Dictionary Fifth Edition, and 'relieving the intensive amount of data processing associated with a specific application from the CPU by performing these calculations in a dedicated or specialized server', web-definition." See Office Action from 07/28/2005, p.3, line 20 to p. 4, line 2. Appellants respectfully traverse and assert that the Examiner has misconstrued the meaning of "off-loading" and is applying the term in a manner inconsistent with the Specification. The Specification and Drawings clearly refer to off-loading of documents (i.e., data files, content) from a document processing system into a repository or an archive and also refer to archiving. See Specification, p. 2, lines 2-3; p. 2, lines 15-17; p. 3, lines 1-3; Fig. 1, element 101; Fig. 2, element 201. Claim 1 recites "[a] method for processing documents being off-loaded in a document processing system...", emphasis added, which is consistent with the Specification and drawings. Off-loading of a document onto a repository or archive is not equivalent to, and cannot be anticipated by, assuming part of the processing demand from another device. The object of the off-loading in the embodiments of the present invention is not processing demand or applications as the Examiner has suggested, but rather documents (data files). Therefore, Appellants respectfully assert that the Examiner must interpret the claim language in view of the Specification and Drawings, and specifically, is in error in interpreting "off-loading" to mean the off-loading of program code, applications, calculations, or data processing for execution by a processor, in a manner inconsistent with the claimed subject matter.

A2. The Examiner's Misinterpretation of Ferguson

Regarding claims 1, 7, and 6, the Examiner recites that:

Ferguson teaches, "providing customized code" (Column 3, lines 47-59 and Column 17 lines 12 – 20 and 59 – 67), "invoking the customized code when the offloading request is issued" (Column 3 line 60 – Column 4 line 35), and "when the post-archiving agent is finished, marking the archived document as 'archived'" (Column 17 lines 30 – 37). See Office Action from 07/28/2005, p.3, lines 8-12.

Appellants respectfully traverse the Examiner's interpretation of the subject matter in *Ferguson*.

Regarding the reference cited by the Examiner as purportedly teaching "providing customized code", in col. 3, lines 47-59, *Ferguson* actually refers to "a number of core features 150 as well as a number of document management utilities as illustrated in Fig. 1B." *Ferguson*, col. 3, lines 47-49. The "core features" in *Ferguson* are Data Storage (STG) Files, *see Ferguson*, col. 3, line 59 to col. 4, line 55; Organization of the Document Collection, *see Ferguson*, col. 4, line 56 to col. 5, line 35; and Virtual Document Storage, *see Ferguson*, col. 5, line 36 to line 64. In addition to these core features, cited as element 150 in Fig. 1B of *Ferguson*, *Ferguson* teaches document management utilities, which are also depicted in Fig. 1B: Indexing and Retrieving, element 157; Categories and Categorization of Documents, element 159; Importing Documents, element 161; Browsing Documents, element 163; Searching Documents, element 167; Viewing Documents, element 169; Clipped Documents, element 171; File Helper (or Archiving), element 173; Directory Monitor, element 175; Task Manager, element 165; Annotations, element 177; and Property Sheets, element 179. Furthermore, col. 17, lines 12-20 in *Ferguson* teaches the description of the File Helper, element 173, with specific functionality with respect to a document collection.

A next utility is the file helper or archiving utility 173. The file helper utility keeps the document collection tidy. More specifically, the file helper utility automatically archives files onto removable media, if, in general, those files have not been accessed or modified for a long period of time. The file helper utility also notifies the user if files have old dates; it notifies the indexing engine and the DCO file when files are taken off-line; it monitors the document collection for document

duplicates; and it organizes a separate index of off-line documents. Ferguson, col. 17, lines 12-20.

The File Helper, element 173, in *Ferguson* thus provides a predefined functionality that is operated using the user-interface illustrated in Figure 18. *Ferguson*, col. 17, lines 21-22. Furthermore, col. 17, lines 59-67 in *Ferguson* teaches the description of the Task Manager, element 165, with specific functionality with respect to a document collection.

The task manager utility 165 is yet another utility employed by the present invention. The task manager utility 165 is a multi-threaded single instance utility that is launched when the host computer is booted after loading the software associated with the present invention or upon a first request for one of its services after the utility has been turned off. Its main function, however, is to facilitate background or batch processing jobs such as importing documents into the document collection. Ferguson, col. 17, lines 59-67.

The Task Manager, element 165, in *Ferguson* thus provides a predefined functionality that is operated using the user-interface illustrated in Figure 21. *Ferguson*, col. 18, lines 1-2. As a result of the foregoing, Appellants respectfully assert that none of the abovementioned elements cited by the Examiner in *Ferguson* are pertinent to "providing customized code" as in claims 1 and 7. The functions cited by the Examiner in *Ferguson* are all disclosed as being operable only with predefined user-interfaces, which is not "providing customized code." There is no teaching or suggestion in *Ferguson* of "providing customized code," which would require providing a programmatic interface to execute a desired functionality.

Regarding the reference cited by the Examiner as purportedly teaching "invoking the customized code when the off-loading request is issued" (as in claims 1 and 7), col. 3, line 60 to col. 4, line 35 in *Ferguson* actually refers to the Data Storage (STG) Files, which is a file created for each document in the document collection. *Ferguson*, col. 3, lines 59-65. In *Ferguson*, the STG file, element 200, is illustrated in Fig. 2A, and does not contain any customized code. There is no teaching or suggestion in *Ferguson* of "invoking customized code when the off-loading request is issued".

Regarding the reference cited by the Examiner as purportedly teaching "when the post-archiving agent is finished, marking the archived document as 'archived' " (as in claim 6), col. 17, lines 30-37 in *Ferguson* actually refers to a secondary user-interface associated with the File Helper, element 173, which *Ferguson* illustrates in Fig. 19. This function in *Ferguson* permits the definition of specific archiving conditions referring to the size or age of a particular document, and discloses maintaining an index to each archived document. However, Appellants respectfully assert that claim 6 limits an agent to, "an agent comprising customized code" (emphasis added), which refers to a programmatic interface, as discussed previously in Section A1 of this paper. Since *Ferguson* does not teach or suggest the operability of a "post-archiving agent," there is no teaching or suggestion in this reference of "when the post-archiving agent is finished, marking the archived document as 'archived' " (emphasis added).

The aforementioned detailed analysis shows, that in her Response, the Examiner has repeatedly cited portions of *Ferguson*, without any support or explanation, that are simply not pertinent to the claim language which they allegedly anticipate. As a result of the foregoing, Appellants respectfully assert that the Examiner has not established that the subject matter in *Ferguson* reads on the claim language in claims 1, 7, and 6, and that these claims are patentable over *Ferguson*.

B. Claim Rejections under 35 U.S.C. §102

Claims 1-7 stand rejected under 35 U.S.C. 102 (e) as anticipated by *Ferguson*. Applicants traverse these rejections.

A claim is anticipated only if every element as set forth in the claim is found in a single prior art reference. MPEP § 2131. An anticipating reference must describe the patented subject matter with sufficient clarity and detail to establish that the subject matter existed and that its existence was recognized by persons of ordinary skill in the field of the invention. *ATD Corp. v. Lydall, Inc.*, 159 F.3d 534,

545 (Fed. Cir. 1998). Further, there must be no difference between the claimed invention and the reference disclosure, as viewed by a person of ordinary skill in the field of the invention. *Scripps Clinic & Research Foundation v. Genentech, Inc.*, 927 F.2d 1565, 1576 (Fed. Cir. 1991). The Examiner has not met this burden as to the claims of the present application.

Claim 1 recites a method for processing documents including, "providing customized code" and "invoking the customized code when an off-loading request is issued." *Ferguson* does not disclose these elements. The Specification of the present application discloses, "LotusNotes supports writing of customized code." See Specification, page 2, line 10. *Ferguson* does not disclose such customized code. The cited portion of *Ferguson* discloses "a number of core features 150 as well as a number of document management utilities." See *Ferguson*, col. 3, lines 47-50. Neither the core features nor the document management utilities are described as "customized code." The Examiner also references *Ferguson's* disclosure of a "file helper utility." See Office Action mailed 07/28/2005 (citing *Ferguson*, col. 17, lines 12-20). *Ferguson's* file helper utility is not "customized code for execution during the off-loading process" nor does *Ferguson* disclose "invoking the customized code when the off-loading request is issued." To the extent the file helper utility may be "customized," the cited portion of *Ferguson* discloses that a user is provided an interface that permits the user to select conditions that trigger the automatic archiving process. See *Ferguson*, col. 17 lines 20-25. Customizing parameters that trigger off-loading is distinguishable from "invoking the customized code when an off-loading request is issued. Clearly, the customized portion of *Ferguson's* file helper utility would have to run before off-loading, since the customized portion is what triggers the off-loading to begin. Therefore, *Ferguson's* disclosure of selecting conditions that trigger an archiving process is distinguishable from claim 1's subject matter relating to "providing customized code" and "invoking the customize code when an off-loading request is issued."

Further, even if *Ferguson's* "file helper utility" is construed as claim 1's "off-loading process," *Ferguson* does not disclose every element of claim 1. *See Ferguson*, col. 17, lines 11-14. *Ferguson's* file helper is described as an "archiving utility" that "keeps the document collection tidy" and "automatically archived files." *See Ferguson*, col. 17, lines 11-50. Therefore, *Ferguson's* "file helper utility" may, for arguments sake, more appropriately be considered claim 1's "off-loading process." However, with this interpretation, *Ferguson* does not disclose the subject matter of claim 1 because *Ferguson* does not disclose "customized code for execution during the off-loading process" and "invoking the customized code when the off-loading request is issued."

The Examiner also cites to *Ferguson's* disclosure of a "task manager utility" as disclosing "customized code for execution during the off-loading process." *See Office Action mailed 07/28/2005* (citing *Ferguson* col. 17, lines 59-67). *Ferguson's* task manager utility is not described as customized code nor is it described as being invoked when an off-loading request is issued. Rather, the task manager utility is launched when a host computer is booted. *See Ferguson*, col. 17, lines 60-63. The task manager utility is described as mainly functioning to facilitate background or batch processing jobs such as importing documents into the document collection. *See Ferguson*, col. 17, lines 63-65. Facilitating batch processing jobs "such as importing documents" is patentably distinguishable from invoking the customized code when an off-loading request is issued. *See Ferguson*, col. 17, lines 64-67. Further, *Ferguson's* task manager utility is neither disclosed as "customized code" nor as being invoked when an off-loading request is issued." Therefore, under the various interpretations of *Ferguson*, claim 1 does not disclose every element of claim 1. Accordingly, claim 1 is patentable over *Ferguson*.

Claim 2 depends from claim 1 and further recites, "the customized code is invoked synchronously to process the document before the document has been off-loaded from the document processing system (emphasis added)." The Examiner rejects claim 2 based in part on *Ferguson's* disclosure of the "file helper utility." *See*

Office Action mailed 07/28/2005 (citing *Ferguson* col. 17, lines 25-33). The cited portion of *Ferguson* discloses that the file helper utility prompts the user before the system archives a document in accordance with the user selected options. *See Ferguson*, col. 17, lines 25-28. *Ferguson's prompting a user before a system archives a document is distinguishable from processing a document before the document has been off-loaded.* Further, as discussed with regard to claim 1, *Ferguson's* file helper utility is not invoked when an off-loading request is issued. Rather, the file helper utility is described as providing a user the ability to select conditions that trigger the off-loading process. Also, since *Ferguson's* file helper is disclosed as the utility which "archives files onto removable media," it should not be construed as "customize code" which is invoked synchronously to the off-loading process. In other words, *Ferguson's* file helper utility should not be construed as disclosing both claim 1's "off-loading process" and claim 1's "customized code for execution during the off-loading process." In summary, *Ferguson* neither discloses every element of claim 1 or 2. Therefore, claim 2 is patentable over *Ferguson*.

Claim 3 depends from claim 1 and further recites, "the customized code is invoked synchronously to process the document after the document has been off-loaded (emphasis added)." The cited portions of *Ferguson* disclose a "file helper utility" that archives files. *See Ferguson*, col. 17, lines 10-12. The cited portions also disclose that the file helper utility prompts a user before the system archives a document. *See Ferguson*, col. 17, lines 25-27. The "file helper utility" is described as maintaining an index of archive documents and storing a thumbnail representation of each archive document. *See Ferguson*, col. 17, lines 33-37. A search engine is capable of searching the content of the thumbnail representations of each archive document. *See Ferguson*, col. 17, lines 37-40. The cited portions do not disclose claim 1's "customized code" that is "invoked synchronously to process the document after the document has been off-loaded." Further, the Examiner has not identified with specificity which elements in *Ferguson* correspond to these elements of claim 1. Searching the content of thumbnail representations and/or STG file data fields is

distinguishable from "process[ing] the document." Therefore, in addition to the reasons discussed above with respect to claim 1, claim 3 is patentable over *Ferguson*.

Claim 4 depends from claim 1 and further recites, "the customized code forms an agent, the agent being invoked via a plug-in interface to the document processing system." The Examiner cites to 25 lines of *Ferguson* without specifying which of *Ferguson's* elements correspond to claim 4's elements. *See* Office Action mailed a 2/25/2004, page 6. For example, the Examiner has not identified which of the cited text discloses "the agent," "the customized code," "the document processing system," or the "plug-in interface." *See* Office Action mailed 02/25/2005, page 6 (citing *Ferguson*, col. 3, lines 30-55). *Ferguson* does not disclose customized code for execution during the off-loading process that is invoked when an off-loading request is issued (claim 1) and that forms an agent invoked via a plug-in interface (claim 4). This claim language refers to interfaces that are inherently programmatic in nature, but *Ferguson* does not teach or suggest a programmatic interface for adding functionality to a document processing system. Therefore, for at least the reasons discussed with regard to claim 1, claim 4 is patentable over *Ferguson*.

Claim 5 depends from claim 1 and further recites, "wherein the customized code runs inside a plug-in architecture." Again, the Examiner cites to 25 lines of *Ferguson* without specifying which of *Ferguson's* elements correspond to claim 5's elements. *See* Office Action mailed a 2/25/2004, page 6. For example, the Examiner has not identified which part of the cited text discloses "the customized code" or "a plug-in architecture." *See* Office Action mailed 02/25/2005, page 6 (citing *Ferguson*, col. 3, lines 30-55). *Ferguson* does not disclose customized code for execution during an off-loading process that is invoked when an off-loading request is issued and that runs inside a plug-in architecture. Therefore, for at least the reasons discussed with regard to claim 1, claim 5 is patentable over *Ferguson*.

Claim 6 is a method for executing an agent comprising customized code. The method includes selecting a document, creating an archiving request to an archiving

engine, invoking a pre-archiving agent (if any), invoking a post-archiving agent (if any), and marking the archived document as 'archived' when the post archiving agent is finished." The Examiner has not cited any disclosure of *Ferguson* which teaches executing an agent comprising customized code. Without specifying which agent in *Ferguson* contains customized code, the Examiner cites to Fig. 1B; the SUMMARY, col. 3, line 24 through col. 4, line 35; and col. 17, line 12 through col. 18, line 35. *Ferguson's* Figure 1B discloses utilities for document importing, directory monitor, indexing and retrieving, categorizing, archiving, property sheet, clipping document, annotation, viewing, searching, browsing, and task manager. None of these utilities in *Ferguson's* Figure 1B is described as an agent comprising customized code. Also, *Ferguson's* Summary does not disclose an agent comprising customized code. *Ferguson's* STG file is not disclosed as an agent comprising customized code. See *Ferguson*, col. 3, lines 60-65. Likewise, instead of being "customized," the STG file and is described as having "a number of standardized data fields" (emphasis added). See *Ferguson*, col. 3, lines 65-67. Similarly, *Ferguson's* file helper utility is not described as containing "customized code." See *Ferguson*, col. 17, lines 12-67. To the extent in *Ferguson's* file helper utility is construed as an agent having "customized code" in that it allows a user to select conditions that trigger automatic archiving, the file helper utility should not also be construed as the "pre-archiving agent."

Further regarding claim 6, *Ferguson's* system would not function as interpreted by the Examiner. According to the Examiner's interpretation, *Ferguson* makes an index of archived documents before the documents are marked "archived." Claim 6, however, requires that archived documents are marked as 'archived,' "when the post-archiving agent is finished" (emphasis added)." The Examiner cites *Ferguson's* disclosure of "the file helper utility continues to make an index of each archived document (acting as a post-archiving agent)." See Office Action mailed 02/25/2005, page 4. The Examiner then cites to *Ferguson's* changing the thumbnail representation as disclosing claim 6's "marking the archived document as archived." Applicants find it illogical that *Ferguson's* system would be able to index "archived"

documents before the documents are marked "archived." Therefore, because *Ferguson's* post-archiving agent (indexing) occurs before *Ferguson's* marking, the Examiner's interpretation of *Ferguson* does not meet the claim elements which require "when the post archiving agent is finished." Therefore, *Ferguson* does not disclose every limitation of claim 6. Accordingly, claim 6 is patentable over *Ferguson*.

Claim 7 recites a system for processing documents being off-loaded, a means for providing customized code for execution during the off-loading process, and a means for invoking the customized code when an off-loading request is issued. The Examiner has not cited any portion of *Ferguson* that discloses a means for providing customized code. The Examiner cites *Ferguson's* disclosure of numerous "core features." See Office Action mailed 5/2005 (citing col. 3, line 47-59). *Ferguson's* core features are not described as a means to provide customized code. The Examiner also cites to *Ferguson* "file helper or archiving utility." The cited portion describes the "file helper utility" keeping the document collection tidy and automatically archiving files onto media. See *Ferguson*, col. 17, lines 1-15. The Examiner also cites to *Ferguson* "task manager utility." The task manager utility is described as a "single instant utility" that is launched when the host computer is booted after loading the software associated with the present invention. See *Ferguson*, col. 17, lines 57-62. The "task manager utility" is not described as a means for providing customized code. Therefore, *Ferguson* does not disclose every limitation of claim 7. Accordingly, claim 7 is patentable over *Ferguson*.

VIII. CONCLUSION

For the reasons noted above, the rejections of claims 1-7 are in error. Appellants respectfully request reversal of the rejections and allowance of claims 1-7.

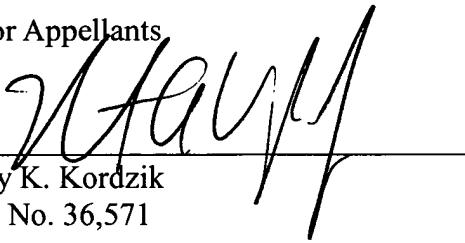
Respectfully submitted,

WINSTEAD SECHREST & MINICK P.C.

Attorneys for Appellants

By:

Kelly K. Kordzik
Reg. No. 36,571


Robert A. Voigt, Jr.
Reg. No. 47,159

P.O. Box 50784
Dallas, Texas 75201
(512) 370-2832

CLAIMS APPENDIX

1. A method for processing documents being off-loaded in a document processing system in which requests are executed by an off-loading process, said method comprising the steps of:
 - providing customized code for execution during the off-loading process; and
 - invoking the customized code when an off-loading request is issued relative to a document in the document processing system synchronously to the off-loading process.
2. The method according to Claim 1, wherein the customized code is invoked synchronously to process the document before the document has been off-loaded from the document processing system.
3. The method according to Claim 1, wherein the customized code is invoked synchronously to process the document after the document has been off-loaded from the document processing system.
4. The method according to Claim 1, wherein the customized code forms an agent, the agent being invoked via a plug-in interface to the document processing system.
5. The method according to Claim 1, wherein the customized code runs inside a plug-in architecture.
6. A method for executing an agent comprising customized code relative to documents being archived from a document processing system, said method comprising the steps of:
 - selecting a document resident in the document processing system for archiving;
 - creating an archiving request for the selected document to an archiving engine associated with the document processing system;
 - invoking a pre-archiving agent, if any, on the selected document;
 - when the pre-archiving agent is finished, archiving the selected document;

invoking a post-archiving agent, if any, on the archived document;
and
when the post-archiving agent is finished, marking the archived document as
'archived'.

7. In a computing environment, a system for processing documents being off-loaded in a document processing system in which requests are executed by an off-loading process, said system comprising:

means for providing customized code for execution during the off-loading process; and
means for invoking the customized code when an off-loading request is issued relative to a document in the document processing system synchronously to the off-loading process.

EVIDENCE APPENDIX

No evidence was submitted pursuant to §§1.130, 1.131, or 1.132 of 37 C.F.R. or of any other evidence entered by the Examiner and relied upon by Appellants in the Appeal.

RELATED PROCEEDINGS APPENDIX

There are no related proceedings to the current proceeding.